WHAT IS CLAIMED IS:

1. A method for managing a scene graph, the method comprising:

   determining a current location for a viewpoint;

5   determining a current locality threshold based on at least the current location of the viewpoint;

   determining which portions of the scene graph are relevant to the current locality threshold;

   loading into a local memory those portions of the scene graph that are relevant within the

10   current locality threshold; and

   replacing portions of the scene graph that are not relevant within the current locality threshold with one or more pointers, wherein the pointers indicate where the replaced portions may be loaded from if the replaced portions are needed.

15   2. The method of claim 1, further comprising:

   determining a predicted future locality threshold; and

   loading into the local memory those portions of the scene graph that are relevant within the predicted future locality threshold, wherein said replacing is performed only on portions of the scene graph that are not relevant within (i) the current locality

20   threshold and (ii) the predicted future locality threshold.

3. The method of claim 2, further comprising:

   determining a current velocity for the viewpoint, wherein the predicted future locality threshold is determined based on at least the current location for the viewpoint and

25   the current velocity of the viewpoint.

4. The method of claim 3, wherein said velocity comprises both translational and rotational components.

5. The method of claim 3, further comprising:

determining a current acceleration for the viewpoint, wherein the predicted future locality

    threshold is determined based on at least the following:

        the current location for the viewpoint,

5   the current velocity of the viewpoint, and

        the current acceleration of the viewpoint.

6. The method of claim 5, wherein said acceleration comprises both translational and

    rotational components.

10

7. The method of claim 3, further comprising:

determining a current orientation for the viewpoint, wherein the predicted future locality

    threshold is determined based on at least the following:

        the current location for the viewpoint,

15   the current velocity of the viewpoint, and

        the current orientation of the viewpoint.

8. The method of claim 3, further comprising caching the replaced portions of the scene

    graph to a local hard drive, wherein the pointers point the replaced portions on the

20   local hard drive.

9. The method of claim 3, wherein the pointers point to network locations from which

    the replaced portions of the scene graph may be loaded.

25 10. The method of claim 3, further comprising:

compressing the replaced portions of the scene graph; and

storing the compressed portions of the scene graph, wherein the pointers indicate where

    the compression portions of the scene graph are stored.

11. The method of claim 1, wherein said replacing is performed only once a predetermined level of memory utilization is reached.

5   12. The method of claim 1, wherein said replacing employs hysteresis to prevent thrashing.

13. A method for managing a scene graph, the method comprising:

determining a current viewpoint for viewing a virtual world of objects described by the

10   scene graph;

determining a predicted future viewpoint for viewing the virtual world;

determining which of the objects in the virtual world are not visible from the current

viewpoint and the predicted future viewpoint; and

replacing portions of the scene graph with pointers, wherein the replaced portions

15   correspond to the objects that are not visible from the current viewpoint and the

predicted future viewpoint, and wherein the pointers indicate where the replaced

portions may be loaded from.

14. The method of claim 13, further comprising:

20   loading any portions of the scene graph that were replaced by pointers and that

correspond to objects that are visible from the current viewpoint or the predicted

future viewpoint; and

removing the pointers that correspond to the loaded portions of the scene graph.

25   15. The method of claim 13, further comprising determining a current velocity for the

current viewpoint, wherein said predicted future locality threshold volume is

calculated based on at least the current viewpoint and the current velocity.

16. The method of claim 15, wherein said velocity comprises both translational and rotational components.

17. The method of claim 13, further comprising determining a current acceleration for the current viewpoint, wherein said predicted future locality threshold volume is calculated based on at least the current viewpoint and the current acceleration.

18. The method of claim 17, wherein said acceleration comprises both translational and rotational components.

19. The method of claim 13, further comprising:

compressing the replaced portions of the scene graph; and

storing the compressed portions of the scene graph, wherein the pointers indicate where the compression portions of the scene graph are stored.

20. A method for managing a scene graph comprising a plurality of graphical objects, the method comprising:

(a) determining a current viewpoint;

(b) determining a predicted future viewpoint;

(c) determining which of the objects are more than a predetermined threshold distance from the current viewpoint and the predicted future viewpoint; and

(d) replacing portions of the scene graph with pointers, wherein the replaced portions correspond to objects that are more than the predetermined threshold distance from the current viewpoint and the predicted future viewpoint, and wherein the pointers indicate where the replaced portions of the scene graph may be loaded from.

21. The method of claim 20, further comprising:

caching the replaced portions of the scene graph to storage locations pointed to by the corresponding pointers.

22. The method of claim 20, further comprising:

5   compressing the replaced portions of the scene graph; and
     storing the compressed portions of the scene graph.

23. The method of claim 20, further comprising:

repeating (a), (b), (c), and (d) as the viewpoint moves or the scene graph changes; and

10   restoring the replaced portions of the scene graph in response to one or more objects
         corresponding to the replaced portions being within the predetermined threshold
         distance from either the current viewpoint or the predicted future viewpoint.

24. The method of claim 20, further comprising:

15   determining a current velocity for the current viewpoint; and
     using the current velocity to determine the predicted future v

25. The method of claim 20, further comprising:

determining a current acceleration for the current viewpoint; and

20   using the acceleration value to determine the predicted future locality threshold.

26. The method of claim 20, wherein the replaced portions of the scene graph may
         comprise visible and non-visible objects.

25   27. A method for managing a scene graph comprising a plurality of graphical objects, the
         method comprising:
     (a) determining a current viewpoint;
     (b) determining a predicted future viewpoint;

(c) determining which of the objects are more than a predetermined threshold distance from the current viewpoint and the predicted future viewpoint; and

(d) replacing one or more of the objects of the scene graph with pointers, wherein the replaced objects are more than the predetermined threshold distance from the current

5   viewpoint and the predicted future viewpoint, and wherein the pointers indicate where the replaced objects of the scene graph may be loaded from.

28. The method of claim 27, further comprising:

caching the replaced objects of the scene graph to storage locations pointed to by the

10   corresponding pointers.

29. The method of claim 27, further comprising:

compressing the replaced objects of the scene graph; and

storing the compressed objects of the scene graph.

15

30. The method of claim 27, further comprising:

repeating (a), (b), (c), and (d) as the viewpoint moves or the scene graph changes; and

restoring the replaced objects of the scene graph in response to one or more of the objects

being within the predetermined threshold distance from either the current viewpoint

20   or the predicted future viewpoint.

31. The method of claim 27, further comprising:

determining a current velocity for the current viewpoint; and

using the current velocity to determine the predicted future viewpoint.

25

32. The method of claim 27, further comprising:

determining a current acceleration for the current viewpoint; and

using the acceleration value to determine the predicted future viewpoint.

33. A computer program embodied on a computer-readable medium, wherein the
computer program comprises a plurality of instructions that are configured to:
determine a current location for a viewpoint;
determine a current locality threshold based on at least the current location of the

5 viewpoint;
determine which portions of the scene graph are relevant to the current locality threshold;
load into a local memory those portions of the scene graph that are relevant within the
current locality threshold; and
replace portions of the scene graph that are not relevant within the current locality

10 threshold with one or more pointers, wherein the pointers indicate where the replaced
portions may be loaded from if the replaced portions are needed.

34. The computer program of claim 27, wherein the instructions are further configured to:
determining a current acceleration; and

15 using the acceleration value to determine the predicted future viewpoint.

35. The computer program of claim 27, wherein the instructions are further configured to:
render one or more frames based on the scene graph.

20 36. The computer program of claim 27, wherein the instructions are further configured to:
receive user input regarding movement of the viewpoint.

37. The computer program of claim 27, wherein the computer program is an application
programming interface (API).

25
38. The computer program of claim 27, wherein the computer program is a graphics
application.

39. A method for managing a scene graph comprising a plurality of pointers, the method comprising:

determining a current location for a viewpoint;

determining a current locality threshold based on at least the current location of the

5   viewpoint;

determining which of the pointers point to data that is relevant to the current locality threshold;

loading into a local memory the data that is relevant to the current locality threshold;

moving the data that is not relevant to the current locality threshold from the local

10   memory to a new location; and

redirecting pointers in the scene graph that correspond to the moved data to point to the new location.

40. The method of claim 39, wherein the current locality threshold equals a current view

15   frustum.

41. The method of claim 39, wherein the moving is only performed if the local memory reaches a predetermined level of fullness.

20